Appendix D

Language Summary

This appendix describes the **vcc** command and the VAX C/ULTRIX language features.

D.1 The vcc Command

The **vcc** command is an ULTRIX command that compiles one or more VAX C source files into one or more object files. The source file or files compiled into an object module is called the compilation unit.

The **vcc** command has the following form:

vcc [*-options* [*args*]]... *filename*[*.type*] [...*filename*[*.type*]] [*-options* [*args*]]

vcc Command Options:

Option	Description
-B *string*	Finds a substitute compiler, preprocessor, an assembler, and a linker in the files named by string. If string is empty, use a standard backup version.
-b	Does not pass the library file **-lc** to the linker. This is a linker option.
-c	Generates an object file with a .o file extension. The linked, executable module is not generated.
-D *name=def*	Assigns the specified value (*def*) to *name*. The preprocessor interprets this option. If a definition value is not specified, the name is set equal to 1.
-E	Runs the vcc preprocessor. The code is preprocessed, and all preprocessor directives, such as include file statements, are resolved.
-Em	Runs the cpp preprocessor and produces the makefile dependencies.
-f	Enables single-precision, floating-point arithmetic. Double-precision, floating point is the default selection. Procedure arguments are still promoted to double-precision, floating-point format.
-g	Generates additional symbol table information for the dbx debugger.
-I*dir*	Specifies the name of the directory containing the relevant include files. A search for included files whose names do not include a directory specification, occurs in: the directory of the file, the directory named by the **-I** option, and finally in directories contained in a standard list.
-K	Generates a full MAP table. This is a linker option. It may be specified on the **vcc** command line or the linker command line.

Option	Description
-l*x*	Specifies a library to include in the link process. The variable *x* is an abbreviation for the library and path name /lib/libx.a in which *x* is a string. If the library is not found, the linker searches for /usr/local/lib /libx.a. A library search starts when the library name is encountered. As a result, the placement of the **-l** within the **vcc** or linker command line is significant.
-Md	Specifies the double-precision, floating-point type as D_floating. This is the default selection. The linker also receives the **-lc** flag.
-Mg	Specifies the double-precision, floating-point type as G_floating. The linker also receives the **-lcg** flag. If you want to use the math library, with code generated with the **-Mg** option, you must link in the G_ FLOAT version of the library by specifying **-lmg** on the linker or **vcc** command line.
-o	Accepts the specified name as the final output file name. This is a linker option. It may be specified on the **vcc** or linker command line.
-O	Invokes the object code improver. The default selection is to perform object code optimization.
-p,-pg	Prepares object files for profiling. The **-pg** option also invokes a run-time recording mechanism that produces a gmon.out file. This file contains more extensive statistics.
-t [0pal]	Finds only the designated compiler, preprocessor, assembler, and linker in the files whose names are constructed by a **-B** option. In the absence of a **-B** option, these are found in the standard places.
-U*name*	Makes the specified variable undefined within the program. This option is interpreted by the preprocessor.
-v *filename.lis*	Produces the listing file, complete with a cross-reference table and machine code listing.
-V *option*	Compiles the source code using vendor-specific options.
-w	Suppresses compiler warning messages. Error messages are displayed, but warning messages are not.
-Y [option]	Compiles a file for one of the following options: SYSTEM_FIVE, BSD, or POSIX. If a parameter other than SYSTEM_FIVE, BSD, or POSIX is specified, a warning is printed and the **-Y** option is ignored. If no parameter is specified, **-Y** defaults to **-YSYSTEM_FIVE**. If multiple **-Y** options are specified, only the last option takes effect, and no warning message is generated.

D.2 Data-Type Keywords

Type Specifiers:

32-bit signed or unsigned:

int
long
long int
unsigned int
unsigned long
unsigned long int

16-bit signed or unsigned:

short
short int
unsigned short
unsigned short int

8-bit signed or unsigned:

char
unsigned char

F_floating format:

float

D_floating or G_floating format:

double
long float

Aggregate types:

struct
union
variant_struct
variant_union

Enumerated type:

enum

Type of function return value:

void

Type declaration:

typedef

Storage-class specifiers:

auto
register
static
extern
globaldef
globalref
globalvalue

Data-type modifiers:

const
volatile

Storage-class modifiers:

readonly
noshare
_align

D.3 Precedence of Operators

In the following table, the operators are listed from highest precedence to lowest. In the binary operator category, operators appearing on higher lines within the category have a higher precedence than the other binary operators.

Category	Association	Operator
Primary	Left to right	( ) [ ] -> .
Unary	Right to left	! ~ ++ – (*type*) – * & **sizeof**
Binary	Left to right	* / % + – << >> < <= > >= == != & ^ | && ||
Conditional	Right to left	?:
Assignment	Right to left	= += –= *= /= %= > >= < <= &= ^= |=
Comma	Left to right	,

D.4 Statements

Syntax:

[*expression*] ;
identifier : *statement*
{ [*declaration-list*] [*statement-list*] }
case [*constant-expression* | default] : *statement-list*
if (*expression*) *statement* [**else** *statement*]
while (*expression*) *statement*
do *statement* **while** (*expression*)
for ([*expression*] ; [*expression*] ; [*expression*])
statement
switch (*expression*) *statement*
break ;
continue ;
return [*expression*] ;
goto *identifier* ;

D.5 Conversion Rules

Arithmetic Conversion

Any operand of type:	Is converted to:

char	**int**
short	**int**
unsigned char	**unsigned int**
unsigned short	**unsigned int**
float	**double**

If operand type is:	The result and the other operands are:
double	**double**
unsigned	**unsigned**

Otherwise, both operands are:	And the result is:
int	**int**

Function Argument Conversion without Prototypes

Any argument of type:	Is converted to type:
float	**double**
char	**int**
short	**int**
unsigned char	**unsigned int**
unsigned short	**unsigned int**
array	pointer to array
function	pointer to function

D.6 VAX C Escape Sequences

The following table lists the VAX C escape sequences:

Character	Mnemonic	Escape Sequence
newline	NL	\n
horizontal tab	HT	\t
vertical tab	VT	\v
backspace	BS	\b
carriage return	CR	\r
form feed	FF	\f
backslash	\	\\
apostrophe	'	\'
quotes	"	\"
bit pattern	ddd	\ddd or \xddd

Use the form "\ddd" to specify any byte value (usually an ASCII code), where the digits ddd are one to three octal digits. The octal digits are limited to 0 to 7.

D.7 Preprocessor Directives

Syntax:

#define *identifier*[([*param1*, . . . *param2*])] *token-string*
#undef *identifier*
#elif *constant-expression*
#include <*file-path*>

#include "*file-path*"
#if *constant-expression*
#ifdef *identifier*
#ifndef *identifier*
#else
#endif
#[**line**] *constant string*
#[**line**] *constant identifier*
#pragma [no]builtins
#pragma [no]inline
#pragma [no]member_alignment
#pragma [no]standard